White paper
Cisco public

The bridge to possible

Azure Arc-Enabled Kubernetes with Cisco IKS (Intersight Kubernetes Service)

© 2022 Cisco and/or its affiliates. All rights reserved.

Contents

Purpose of this document 3

Introduction 3

Azure Arc enabled Kubernetes 4

Azure Arc enabled Kubernetes Validation Program 5

Connect IKS cluster to Azure with Azure Arc Enabled for Kubernetes 6

Monitor using Azure Monitor for Containers 14

Deploy configurations using GitOps on an Azure Arc-enabled Kubernetes cluster 19

Connect Cisco IKS cluster to Azure using Azure CLI 24

For more information 26

© 2022 Cisco and/or its affiliates. All rights reserved.

Purpose of this document

This document provides step-by-step procedures to connect Cisco Intersight™ Kubernetes Service (IKS) to Microsoft Azure using Azure Arc-enabled Kubernetes.

Introduction

Cisco Intersight (https://intersight.com) is an API-driven, cloud-based, software-as-a-service (SaaS) hybrid cloud operations platform. It delivers intelligent automation, observability, and optimization for traditional and cloud-native applications and infrastructure. It is a new generation of global management tool for the Cisco Unified Computing System™ (Cisco UCS®), Cisco HyperFlex™ systems, other Cisco Intersight-connected devices, and third-party Intersight-connected devices. It provides a holistic and unified approach to managing distributed infrastructure, cloud services, virtualized workloads, and container orchestration platforms. As a result, customers can achieve significant TCO savings and deliver applications faster in support of new business initiatives.

Cisco Intersight simplifies provisioning of servers, associated storage, and fabric automatically with model-based configurations. Along with predefined profiles, IT can align policy for consistent compliance and productivity and can lower the risk of failures. To enhance the customer experience, Cisco Intersight also has a guided wizard to aid in the definition of profiles, set rules, and operating characteristics.

Cisco Intersight Kubernetes Service (IKS) introduces lifecycle management capabilities into the Intersight orchestration platform to offer Kubernetes as a service. It enables customers to quickly provision, deploy, and easily manage the lifecycle of Kubernetes clusters across the globe using a single cloud portal – Cisco Intersight. IKS also has a full stack of observability, monitoring, and logging for Kubernetes management. With IKS, Cisco Intersight delivers a turn-key SaaS solution for deploying and operating consistent, production-grade Kubernetes clusters anywhere.

IKS allows you to create Kubernetes cluster profiles to configure and deploy clusters anywhere. A cluster profile contains information such as the infrastructure provider, the target environment, and the node configuration. You can set these configurations by choosing existing Kubernetes policies or by creating new policies. IKS policies define the configurations that can be used across multiple cluster profiles. Using policies helps you to quickly and easily update multiple cluster profile configurations. Whenever you update a policy, the changes apply to all cluster profiles that use the policy. The Kubernetes clusters can be created by applying the Kubernetes cluster profiles from the Cisco Intersight GUI, or the deployment can be automated using the Terraform provider for Cisco Intersight.

© 2022 Cisco and/or its affiliates. All rights reserved.

Azure Arc is an offering of Azure service that simplifies the governance and management of complex and distributed environments across on-premises, edge, and multicloud. It provides you with a single pane of glass by projecting your non-Azure, on-premises, or other cloud resources (like AWS, GCP, etc.) into Azure Resource Manager. With Azure Arc you can:

- easily organize, govern, and secure servers (Windows and Linux), SQL Servers, and Kubernetes clusters across data centers, the edge, and multicloud environments and use Azure tools like Azure Policy and Azure Resource Graph with both traditional and cloud workloads.
- ensure consistent deployments and configuration – deploy and manage Kubernetes applications with GitHub and Azure Policy. Ensure that applications and clusters are consistently deployed and configured at scale from source control.
- automate and enforce policies to meet data governance and security requirements, and manage costs efficiently. Get the latest cloud innovation and automation, elastic scale, and unified management for data workloads that are running across hybrid infrastructure.

Azure Arc enabled Kubernetes

With Azure Arc enabled Kubernetes, you can attach and configure Kubernetes clusters running anywhere. When the Kubernetes cluster is connected to Azure Arc it will appear in the Azure portal as a resource with an ARM ID and a managed identity placed under Azure subscription and resource group and can receive tags like any other Azure resource. The Kubernetes cluster admin needs to deploy agents to connect the cluster to Azure and these agents run in the 'azure-arc' Kubernetes namespace as standard Kubernetes deployments, handle connectivity to Azure, collect Azure Arc logs and metrics and watch for configuration requests.

Azure Arc-enabled Kubernetes supports the following scenarios:

- Connect Kubernetes running outside of Azure for inventory, grouping, and tagging.
- Deploy applications and apply configuration using GitOps-based configuration management.
- View and monitor your clusters using Azure Monitor for containers.
- Enforce threat protection using Azure Defender for Kubernetes.
- Apply policy definitions using Azure Policy for Kubernetes.
- Create custom locations as target locations for deploying Azure Arc-enabled Data Services, App Services on Azure Arc and Event Grid on Kubernetes.

© 2022 Cisco and/or its affiliates. All rights reserved.

Azure Arc enabled Kubernetes Validation Program

Azure Arc-enabled Kubernetes works with any Cloud Native Computing Foundation (CNCF) certified Kubernetes clusters. The Azure Arc team has worked with key industry partners to validate conformance of their Kubernetes distributions with Azure Arc-enabled Kubernetes.

Cisco IKS participated in the validation program and has successfully passed the conformance tests for Azure Arc-enabled Kubernetes for IKS. For more information on Azure Arc service and technology partners and the validated distributions, refer the following links: https://techcommunity.microsoft.com/t5/azure-arc-blog/azure-arc-service-and-technology-partners/ba-p/2478102

https://docs.microsoft.com/en-us/azure/azure-arc/kubernetes/validation-program

The conformance tests run as part of the Azure Arc-enabled Kubernetes validation cover the following scenarios:

1. Connect Kubernetes clusters to Azure Arc:

- Deploy Azure Arc-enabled Kubernetes agent Helm chart on cluster.
- Set up Managed System Identity (MSI) certificate on cluster.
- Agents send cluster metadata to Azure.

2. Configuration:

- Create configuration on top of Azure Arc-enabled Kubernetes resource.
- Flux, needed for setting up GitOps workflow, is deployed on the cluster.
- Flux pulls manifests and Helm charts from demo Git repo and deploys to cluster.

© 2022 Cisco and/or its affiliates. All rights reserved.

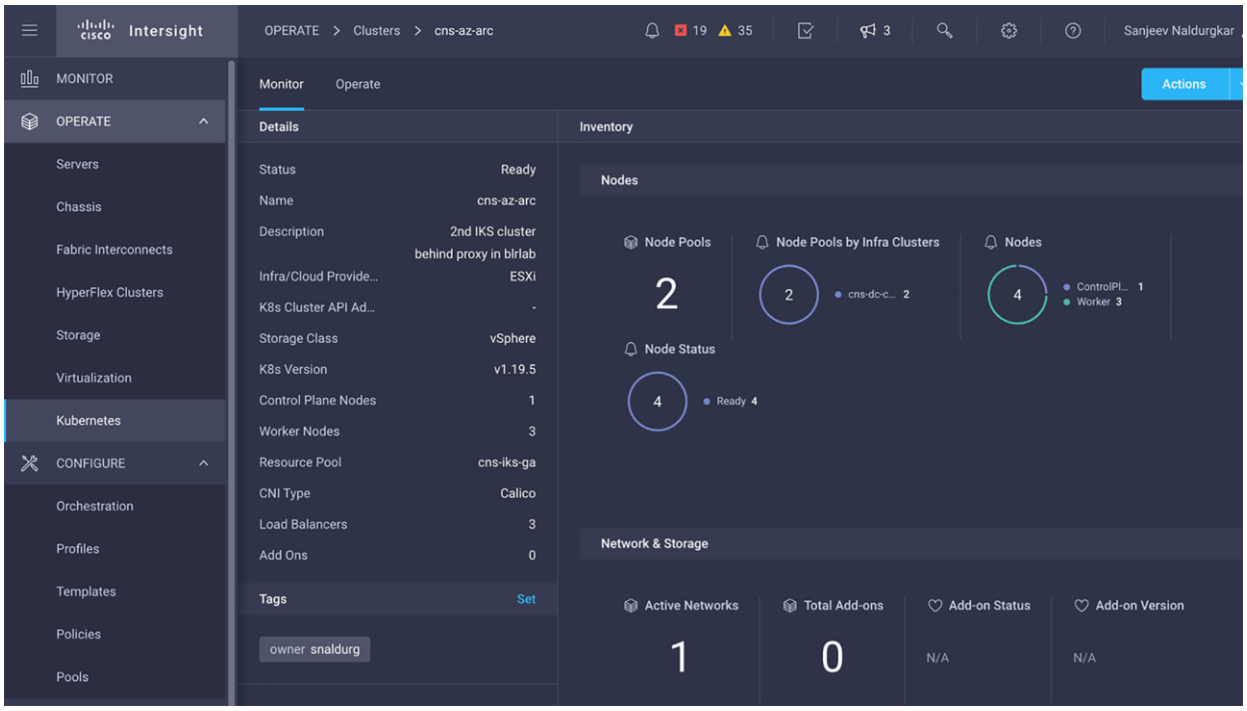
Connect IKS cluster to Azure with Azure Arc Enabled for Kubernetes

This section covers the steps to connect a Cisco IKS cluster to Azure.

Prerequisites

To connect/onboard an existing Kubernetes cluster to Azure Arc, make sure the below prerequisites are installed.

1. A Cisco IKS cluster up and running or refer this link to install a new Kubernetes cluster.



© 2022 Cisco and/or its affiliates. All rights reserved.

2. An Azure subscription with “read' and 'write' permissions on the Azure Arc-enabled Kubernetes resource type (Microsoft.Kubernetes/connectedClusters).
3. Setup your local machine/workstation with the below tools and extensions:

- Install Azure CLI with version >=2.16.0.

```
snaldurg@DemoVM-VirtualBox:~$ az --version
azure-cli                         2.29.0 *

core                              2.29.0 *
telemetry                          1.0.6

Extensions:
connectedk8s                       1.2.0
k8s-configuration                  1.1.1
k8s-extension                      0.7.1
customlocation                     0.1.3
resource-graph                     2.1.0
arcdata                            1.1.1

Python location '/opt/az/bin/python3'
Extensions directory '/home/snaldurg/.azure/cliextensions'

Python (Linux) 3.6.10 (default, Oct  8 2021, 09:26:22)
[GCC 9.3.0]

Legal docs and information: aka.ms/AzureCliLegal
```

- Install the latest Azure CLI extensions – connectedk8s, k8s-configuration, k8s-extension and custom location.

```
> az extension add --name connectedk8s
> az extension add --name k8s-configuration
> az extension add --name k8s-extension
> az extension add --name customlocation
> az extension list -o table
```

```
snaldurg@DemoVM-VirtualBox:~$ az extension list -o table
Experimental    ExtensionType    Name              Path                                                          Preview    Version
--------------  ---------------  ----------------  ------------------------------------------------------------  ---------  ---------
False           whl              connectedk8s      /home/snaldurg/.azure/cliextensions/connectedk8s              False      1.2.0
False           whl              k8s-configuration /home/snaldurg/.azure/cliextensions/k8s-configuration         False      1.1.1
False           whl              k8s-extension     /home/snaldurg/.azure/cliextensions/k8s-extension            True       0.7.1
False           whl              customlocation    /home/snaldurg/.azure/cliextensions/customlocation           False      0.1.3
```

- A kubeconfig file with cluster admin permissions.

```
> kubectl cluster-info
```

```
snaldurg@DemoVM-VirtualBox:~$ kubectl cluster-info  --kubeconfig ./cns-az-arc-kubeconfig.yml
Kubernetes control plane is running at https://10.127.61.77:6443
KubeDNS is running at https://10.127.61.77:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
snaldurg@DemoVM-VirtualBox:~$
```

© 2022 Cisco and/or its affiliates. All rights reserved.

- Install Helm 3.

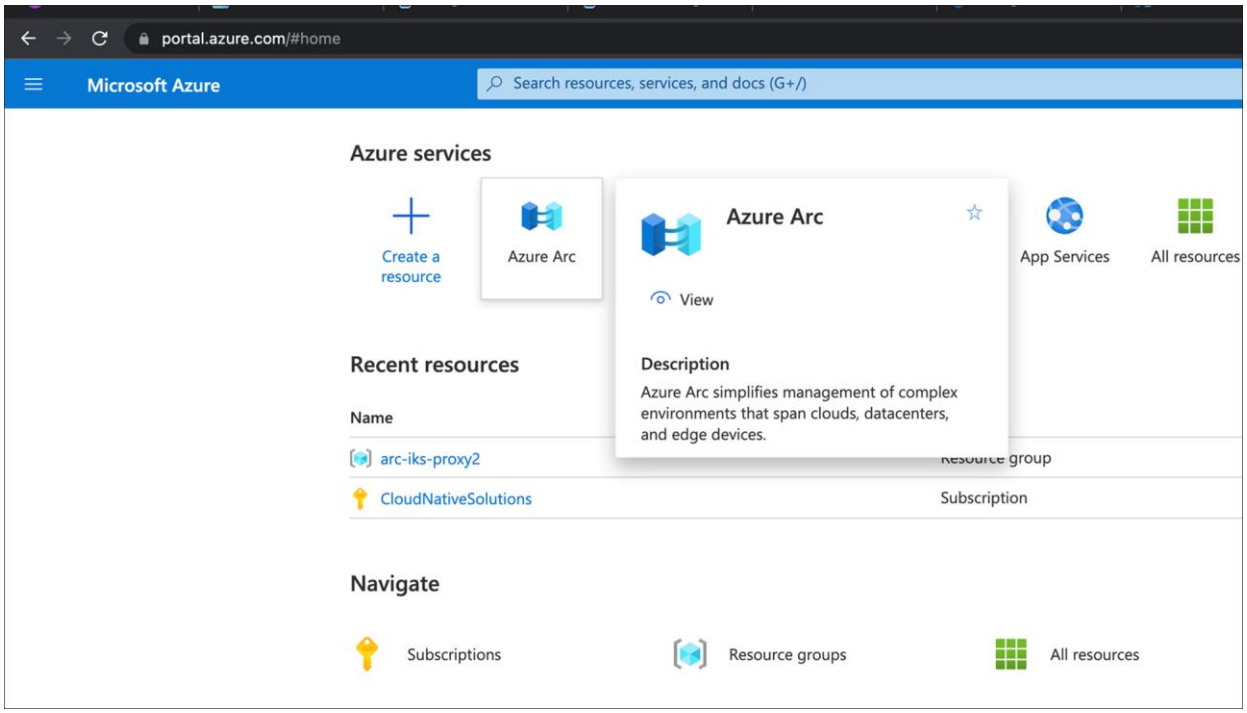
  ```
  > helm version
  ```

```
snaldurg@DemoVM-VirtualBox:~$ helm version
version.BuildInfo{Version:"v3.7.1", GitCommit:"1d11fcb5d3f3bf00dbe6fe31b8412839a96b3dc4", GitTreeState:"clean
", GoVersion:"go1.16.9"}
snaldurg@DemoVM-VirtualBox:~$
```

- The Kubernetes clusters has access to ports 443 and 9418 and the required outbound URLs.

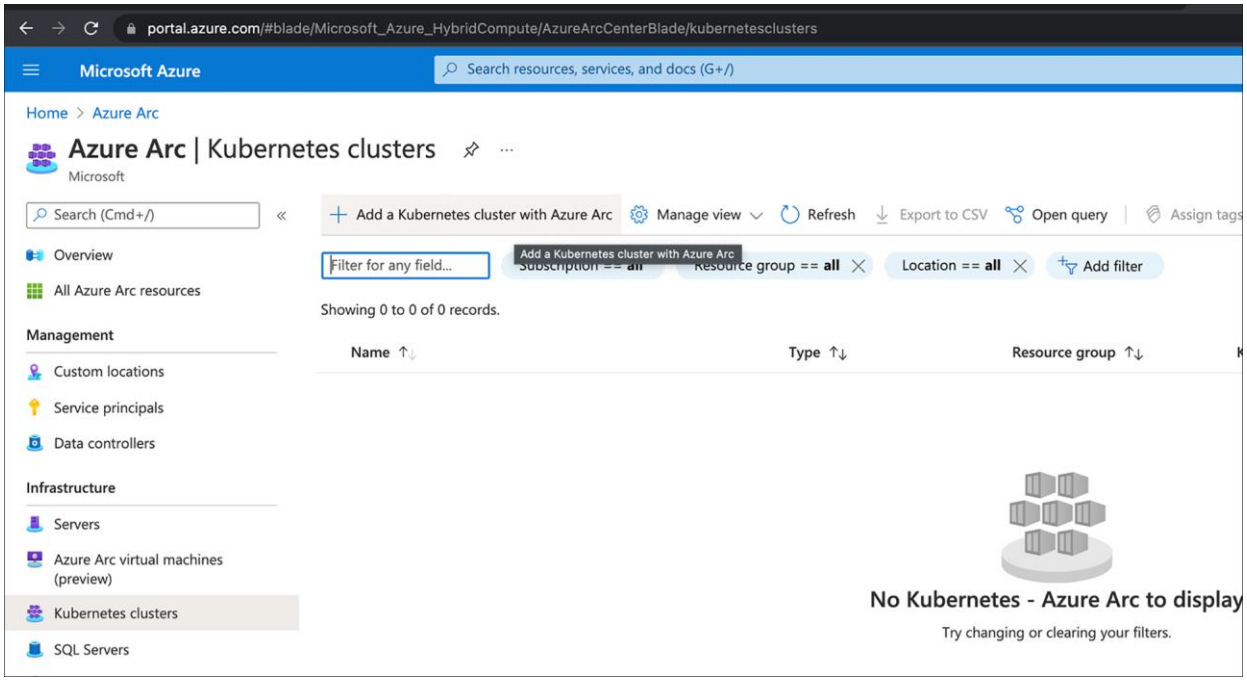
Connect the IKS cluster to Azure

1. Log in to the Azure portal, search for Azure Arc, and click on it to open.



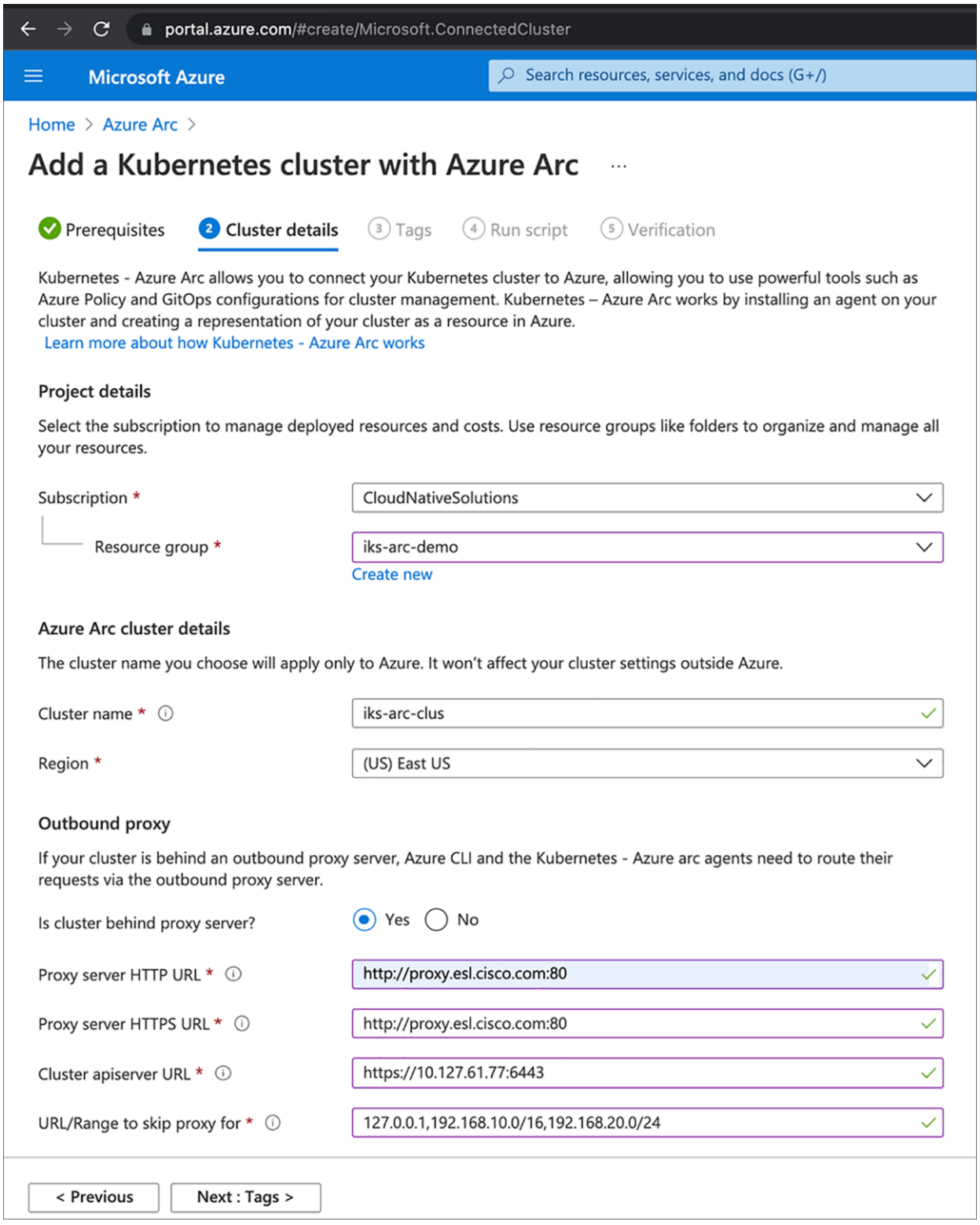
© 2022 Cisco and/or its affiliates. All rights reserved.

2. In the Azure Arc page, select “Kubernetes clusters” from under the Infrastructure section and click on “Add a Kubernetes cluster with Azure Arc”.



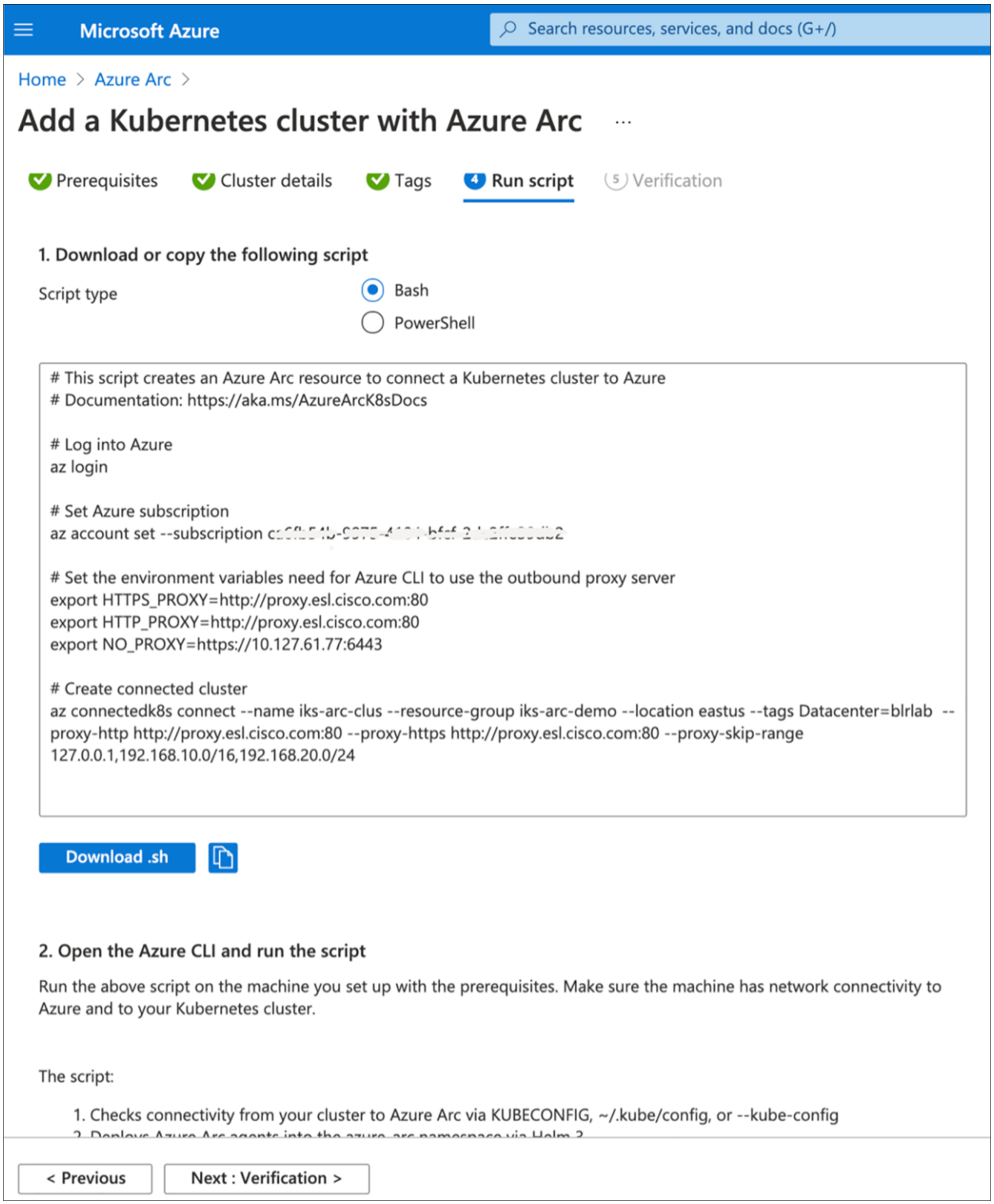
© 2022 Cisco and/or its affiliates. All rights reserved.

3. In the prerequisites page, make sure they are fulfilled before adding the cluster to Azure Arc and click Next.
4. In the Cluster details page, select and fill the details as shown in the below figure and click Next.



© 2022 Cisco and/or its affiliates. All rights reserved.

5. This step is optional. However, you can create/assign tags and click Next.
6. Click on the “Download.sh” tab to download the script to your local machine with prerequisites installed.



© 2022 Cisco and/or its affiliates. All rights reserved.

7. To make the downloaded file executable, enter the below command on your local machine installed with the prerequisites

```
> chmod +x Download.sh
```

```
snaldurg@DemoVM-VirtualBox:~$ chmod +x Download.sh
snaldurg@DemoVM-VirtualBox:~$
```

8. Run the script and as mentioned open the page in a web browser and sign in with the code to authenticate.

```
> ./Download.sh
```

```
snaldurg@DemoVM-VirtualBox:~$ ./Download.sh
To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code H3F23DCP8
 to authenticate.
```

9. Select an Azure account to complete the sign in for the script execution to complete. The below figure shows the output of a successful script execution.

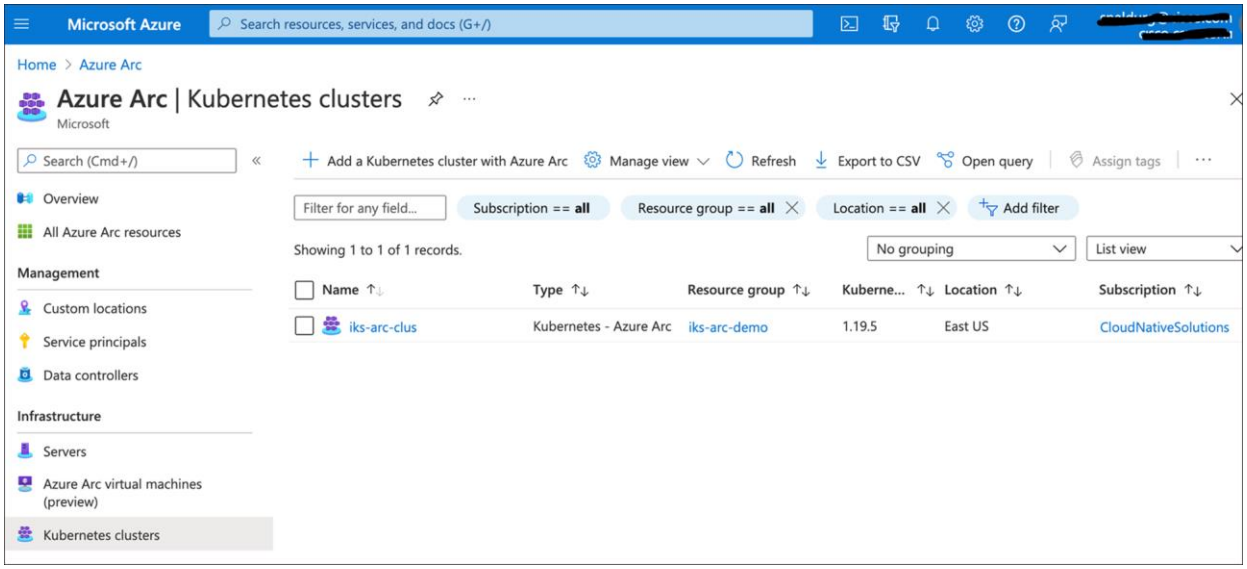
10. Verify the cluster connection using the command below:

```
> az connectedk8s list --resource-group <Resource Group Name> --output table
```

```
snaldurg@DemoVM-VirtualBox:~$ az connectedk8s list --resource-group iks-arc-demo --output table
Name          Location    ResourceGroup
------------  ----------  ---------------
iks-arc-clus  eastus      iks-arc-demo
```

© 2022 Cisco and/or its affiliates. All rights reserved.

11. You can also view the cluster connection listed in the Azure portal by searching for Azure Arc services.



12. View the Azure Arc agents deployed on the Kubernetes cluster using the command below:

```
> kubectl get deployments,pods -n azure-arc
```

```
snaldurg@DemoVM-VirtualBox:~$ kubectl -n azure-arc get deployments,pods
NAME                                        READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/cluster-metadata-operator   1/1     1            1           13m
deployment.apps/clusterconnect-agent        1/1     1            1           13m
deployment.apps/clusteridentityoperator     1/1     1            1           13m
deployment.apps/config-agent                1/1     1            1           13m
deployment.apps/controller-manager          1/1     1            1           13m
deployment.apps/extension-manager           1/1     1            1           13m
deployment.apps/flux-logs-agent             1/1     1            1           13m
deployment.apps/kube-aad-proxy              1/1     1            1           13m
deployment.apps/metrics-agent               1/1     1            1           13m
deployment.apps/resource-sync-agent         1/1     1            1           13m

NAME                                            READY   STATUS    RESTARTS   AGE
pod/cluster-metadata-operator-566cb95f7f-2tjsx  2/2     Running   0          13m
pod/clusterconnect-agent-7d4b6c9d7d-pfqwd       3/3     Running   0          13m
pod/clusteridentityoperator-858846bbd-p8q16     2/2     Running   0          13m
pod/config-agent-66c5f6bddd-kdmxp               2/2     Running   0          13m
pod/controller-manager-76ff8767d9-h8gbx         2/2     Running   0          13m
pod/extension-manager-845bdf48b8-mhbb9          2/2     Running   0          13m
pod/flux-logs-agent-689cccc8d6-nzqqg            1/1     Running   0          13m
pod/kube-aad-proxy-5d4679478-6n49p              2/2     Running   0          13m
pod/metrics-agent-697b7c69cf-8glcr              2/2     Running   0          13m
pod/resource-sync-agent-7579476848-tpdpt        2/2     Running   0          13m
snaldurg@DemoVM-VirtualBox:~$
```

13. Verify the Azure Arc helm status using the command below:

```
> helm --namespace default status azure-arc
```

```
snaldurg@DemoVM-VirtualBox:~$ helm --namespace default status azure-arc
NAME: azure-arc
LAST DEPLOYED: Tue Oct 19 16:31:35 2021
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
snaldurg@DemoVM-VirtualBox:~$
```

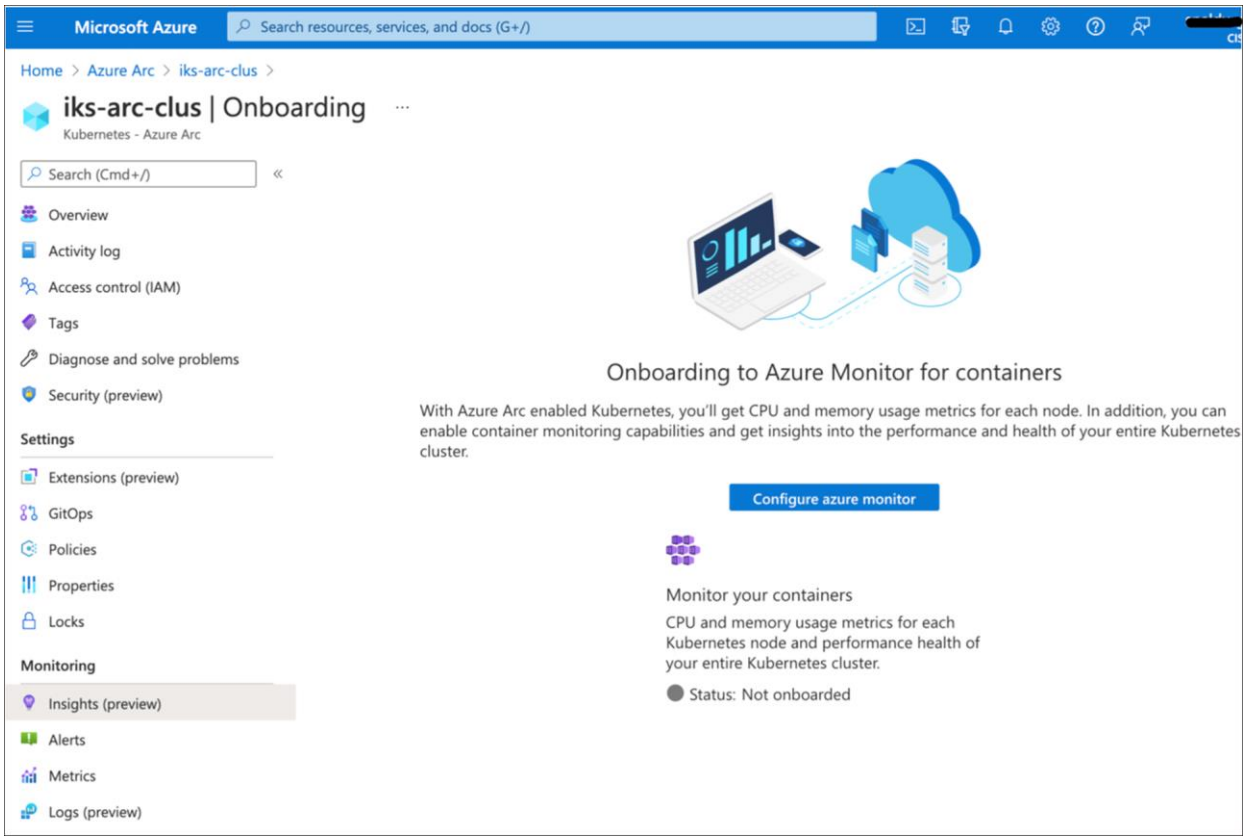
© 2022 Cisco and/or its affiliates. All rights reserved.

Monitor using Azure Monitor for Containers

Azure Monitor Container Insights for Azure Arc-enabled Kubernetes clusters provides rich monitoring experience for Azure Arc-enabled Kubernetes clusters. The container insights feature enables you to understand the performance and health of your Kubernetes cluster and the container workloads.

Follow the below steps to configure Azure Monitor for containers from the Azure Arc-enabled Kubernetes resource blade:

1. In the Azure portal, select the Azure Arc-enabled Kubernetes cluster that you wish to monitor.
2. Select the “Insights” item under the “Monitoring” section of the resource blade.
3. On the onboarding page, select the “Configure Azure Monitor” button.



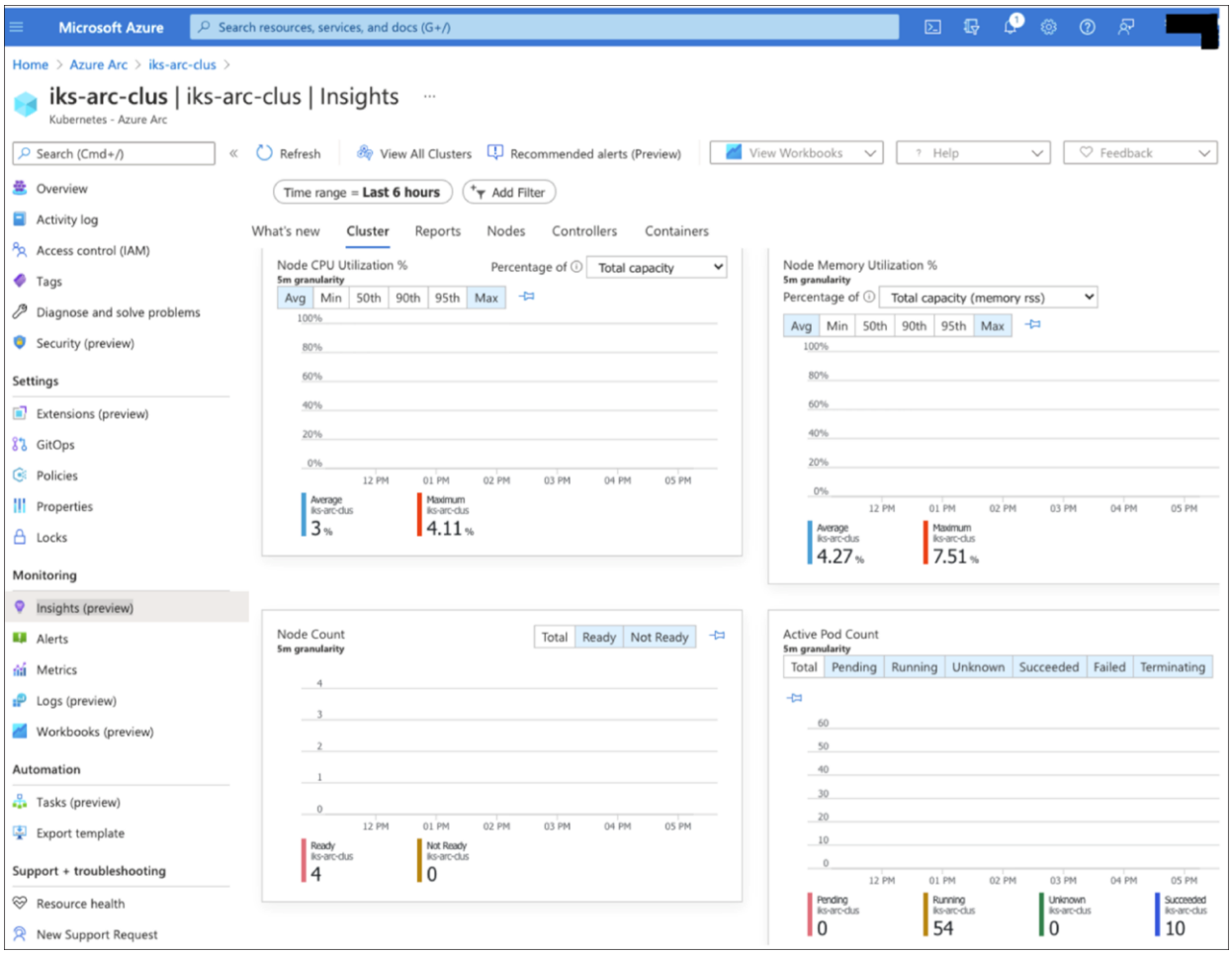
4. You can now choose the Log Analytics workspace to send your metrics and logs data to.
5. Select the “Configure” button to deploy the Azure Monitor Container Insights cluster extension.

© 2022 Cisco and/or its affiliates. All rights reserved.

6. Insights under the Monitoring section provides insights on cluster, reports, nodes, controllers and containers. For example, the below figures shows insights of the cluster, nodes and deployment reports.

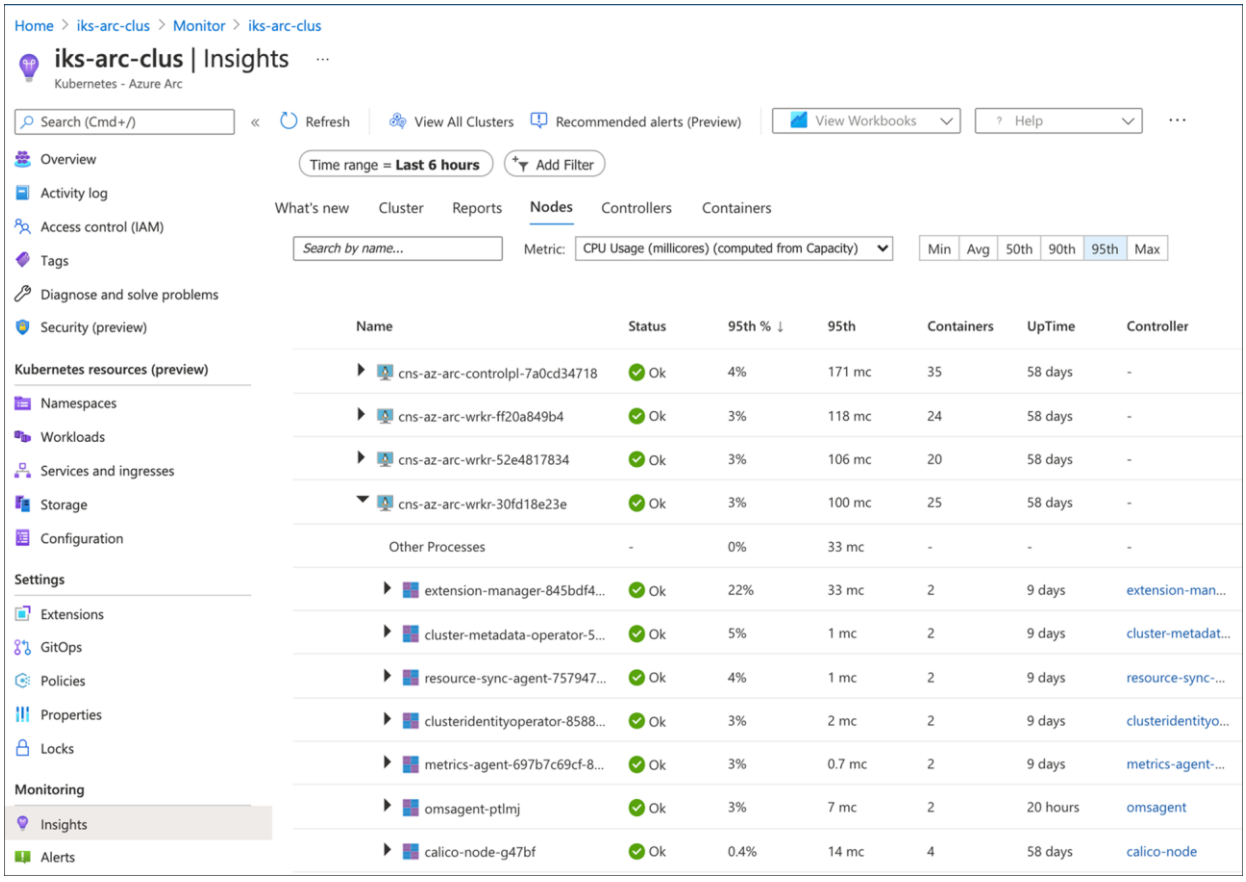
   You can access container insights in two ways from the Azure portal: from Azure Monitor or directly from the Azure Arc–enabled Kubernetes cluster.

   For example, the figure below shows some key performance metrics of your cluster. To access this page, navigate to Home > Azure Arc > Kubernetes Cluster > Monitoring > Insights, and click on the Cluster tab.



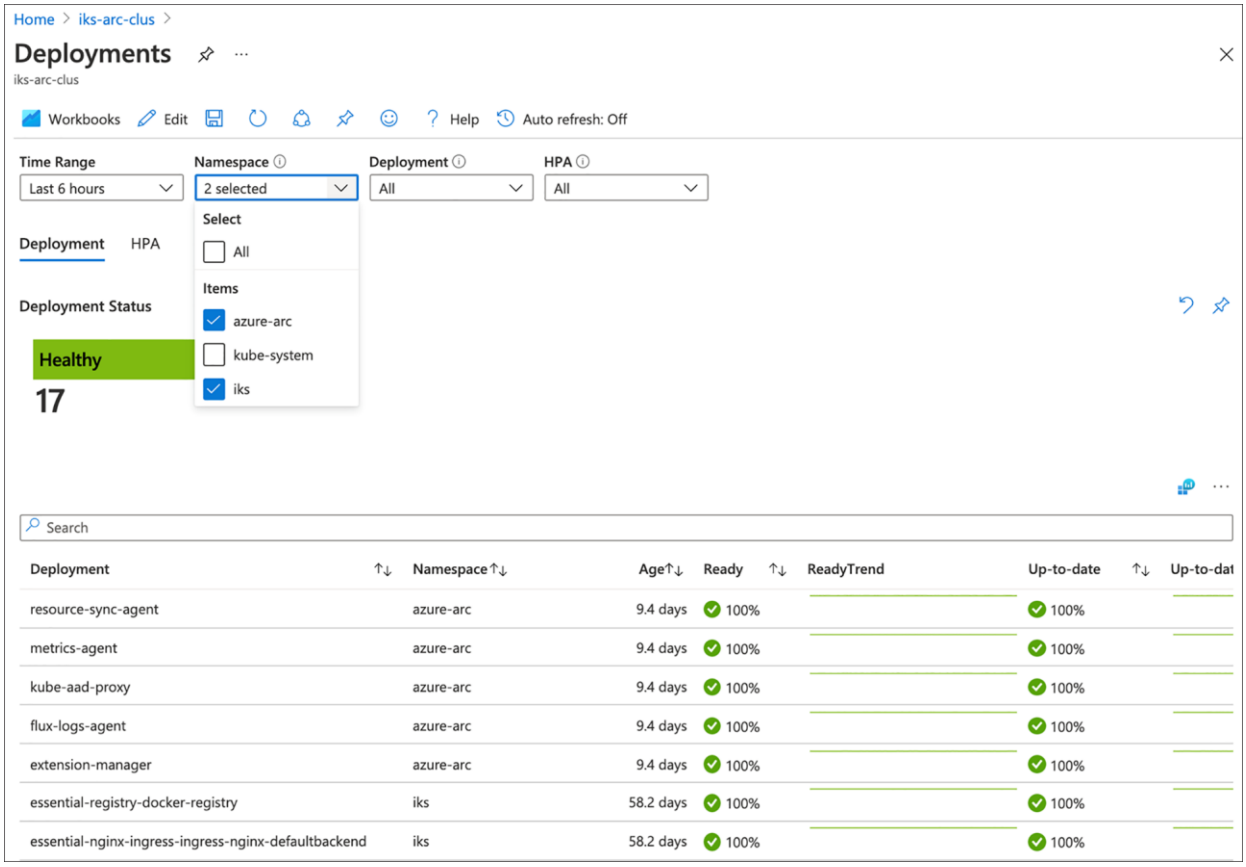
   You can switch to other tabs to display and analyse the health of nodes, controllers, and containers. The figure below shows the status and performance metrics of cluster nodes. You can expand the objects by clicking on the arrow button; selecting an object shows its properties on the right-side of the pane.

© 2022 Cisco and/or its affiliates. All rights reserved.



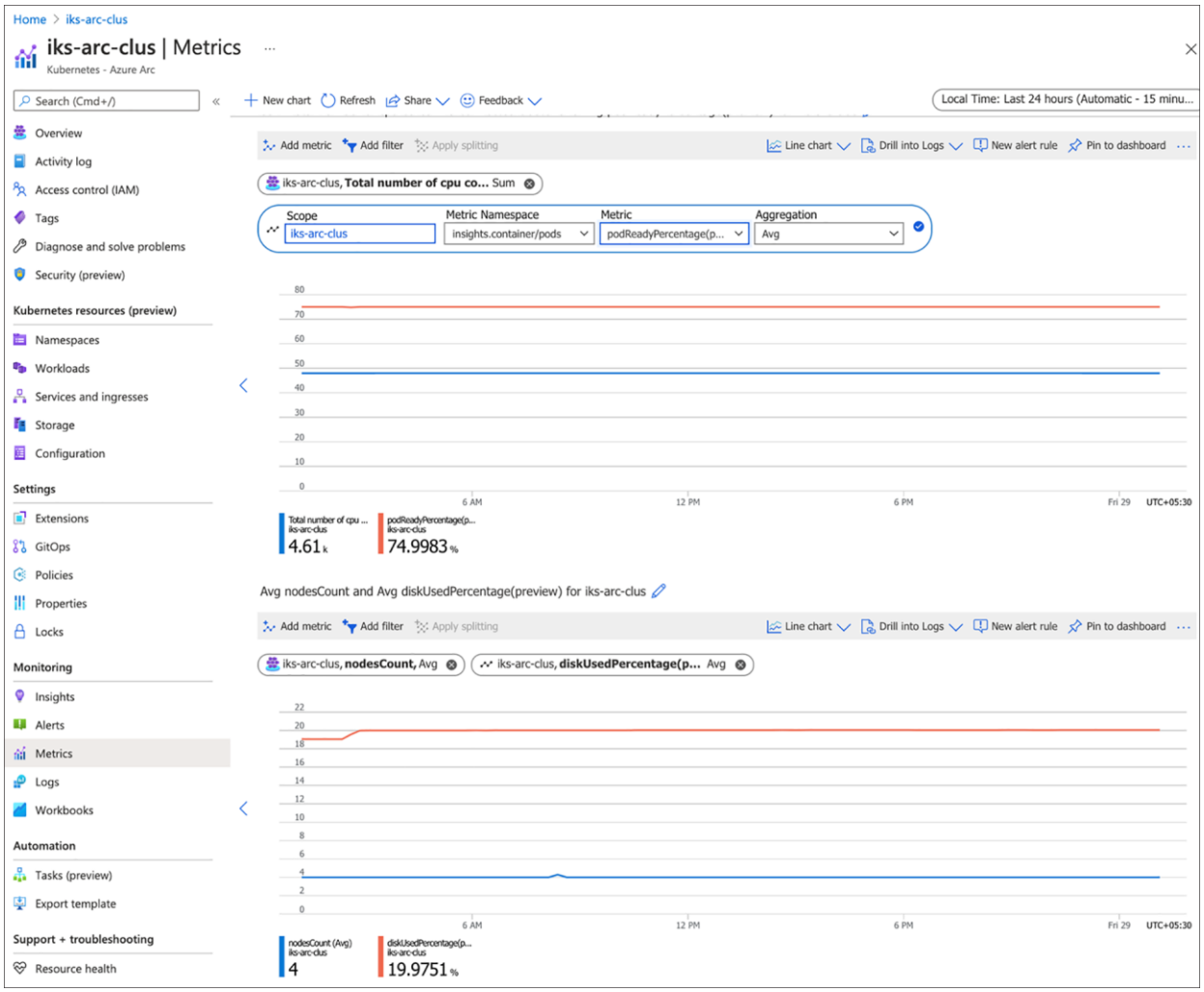
Reports in container insights have some recommended out-of-the-box Azure workbooks. For example, the figure below shows the status and health of the deployments by navigating to Reports > Resource Monitoring workbooks and clicking on Deployments. You can also create a custom workbook.

© 2022 Cisco and/or its affiliates. All rights reserved.



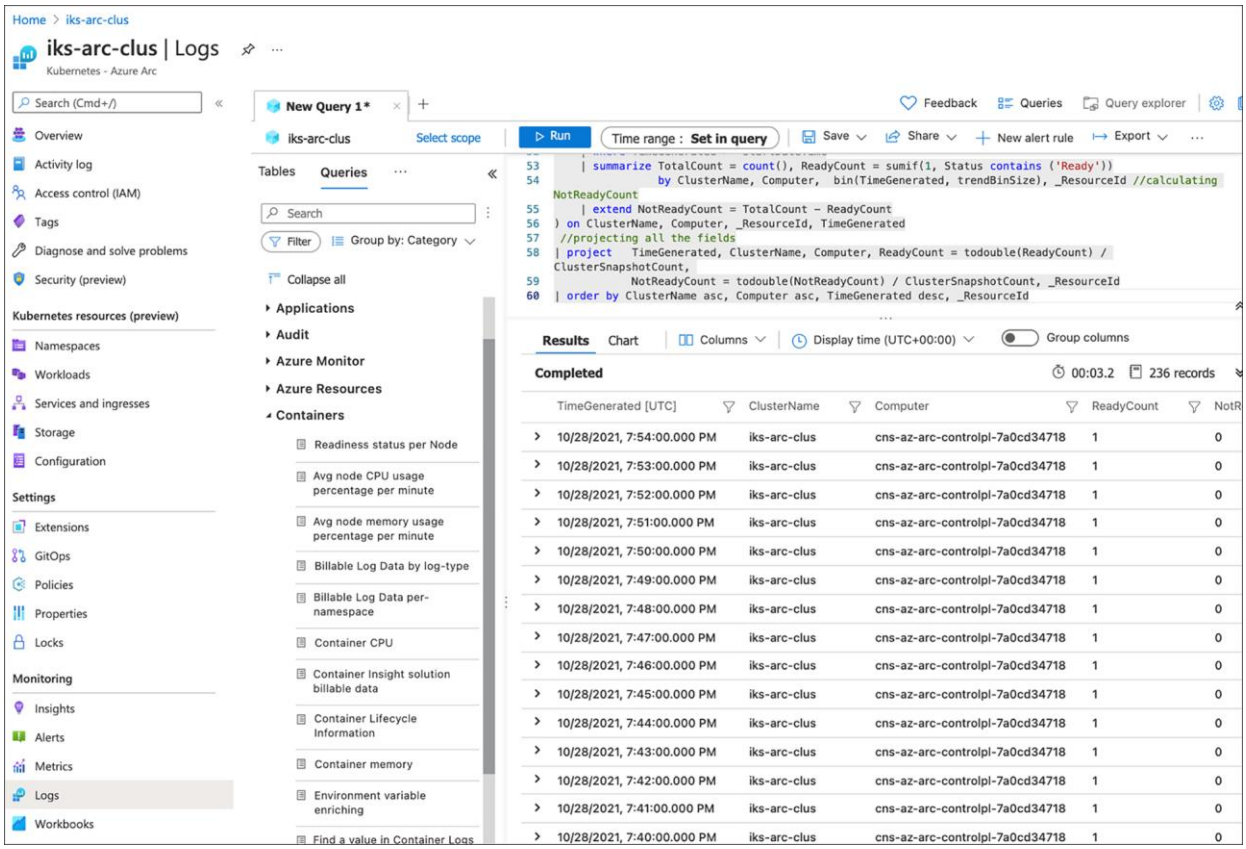
© 2022 Cisco and/or its affiliates. All rights reserved.

7. Azure Monitor Metrics, a feature of Azure Monitor, collects numeric data at regular intervals from resources into a time series database. You can use this feature to analyse, alert, visualize, automate, retrieve, export, and archive based on the data collected. The figure below shows some of the metrics for the connected Kubernetes cluster:



© 2022 Cisco and/or its affiliates. All rights reserved.

8. Azure Monitor Logs is another feature of Azure Monitor that collects and organises log and performance data; it allows data to be analysed using a query language. The figure below shows the logs for a container's readiness status per node.



Deploy configurations using GitOps on an Azure Arc-enabled Kubernetes cluster

This section covers the steps to apply configurations using GitOps on Cisco IKS Azure Arc-enabled Kubernetes cluster. For this document purpose, we are using an example public repository which holds the Kubernetes resources and Helm charts that needs to be applied to the cluster. To associate a private repository with the configuration, refer this URL. The manifests in this repository provision a few namespaces, deploy workloads, and provide some team-specific configuration. Using this repository with GitOps creates the following resources on your cluster:

- Namespaces: cluster-config, team-a, team-b
- Deployment: arc-k8s-demo
- ConfigMap: team-a/endpoints

The config-agent polls Azure for new or updated configurations.

© 2022 Cisco and/or its affiliates. All rights reserved.

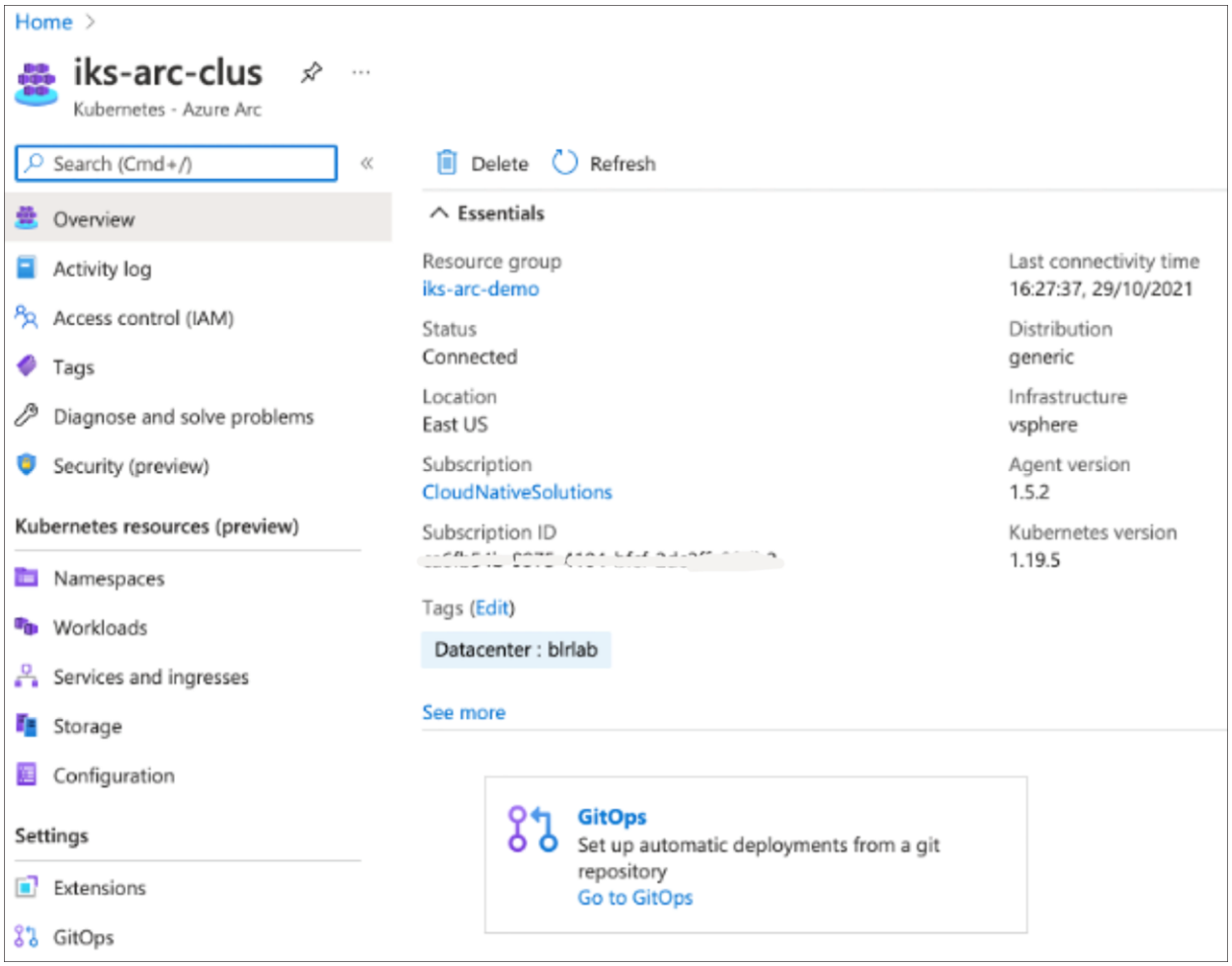
Prerequisites

- An Azure account with an active subscription.
- An existing Azure Arc-enabled Kubernetes connected cluster.
- An understanding of the benefits and architecture of this feature. Read more in [Configurations and GitOps - Azure Arc-enabled Kubernetes article](#).
- Install the k8s-configuration Azure CLI extension of version >= 1.0.0:

```
> az extension add --name k8s-configuration
```

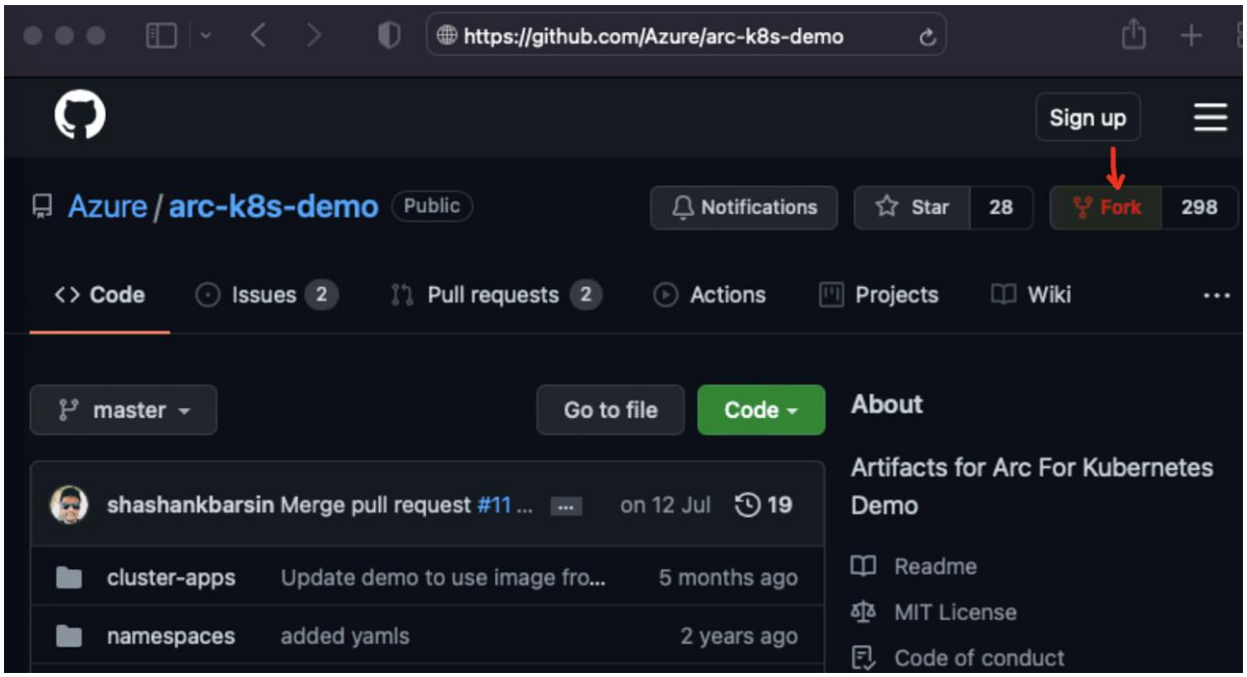
Deploy configurations using GitOps

1. In Azure portal, navigate to Azure Arc > Kubernetes and open a connected IKS cluster on which you want to deploy GitOps configuration.
2. Click on GitOps box seen in the overview section of the cluster or you can also find it under the Settings section on the left pane.

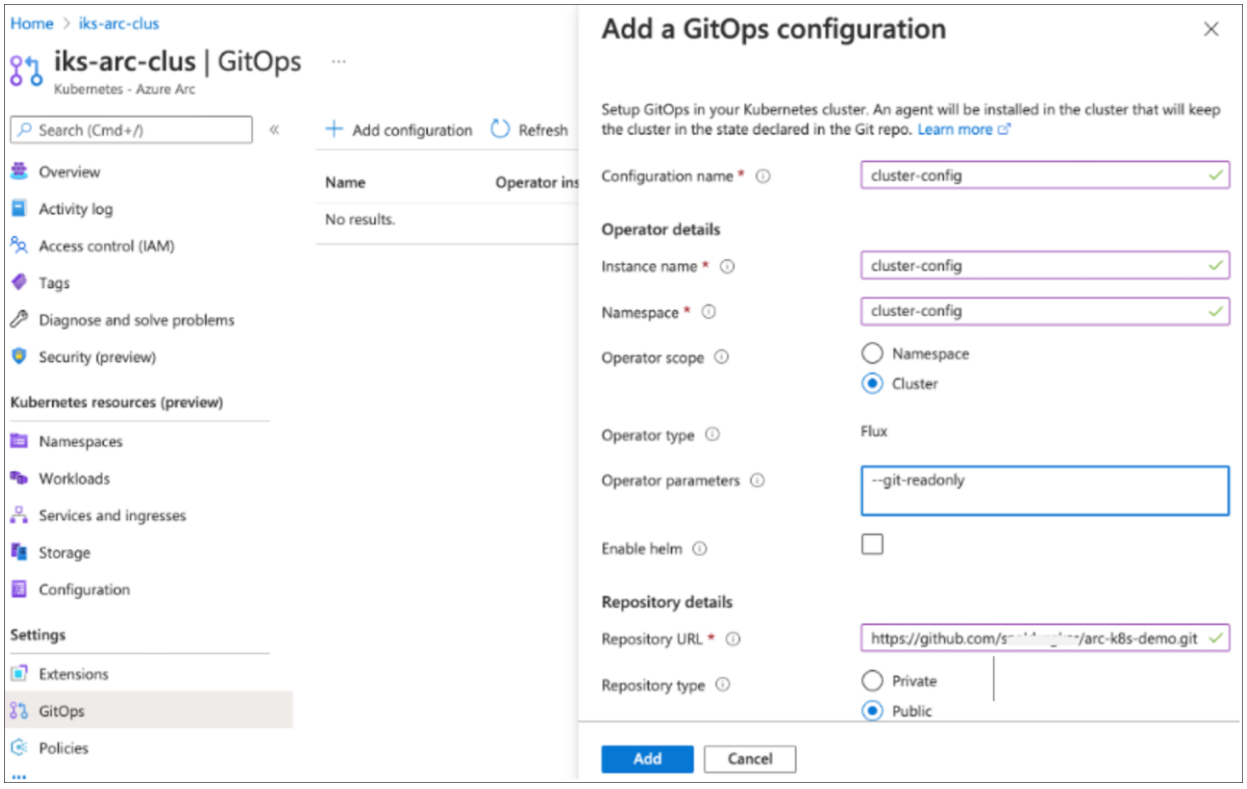


© 2022 Cisco and/or its affiliates. All rights reserved.

3. Click on this github link and fork the repository. This requires the Git client to be installed on the administrator's workstation. This is a public repository containing official cluster configuration for Azure Arc–enabled Kubernetes for demos.

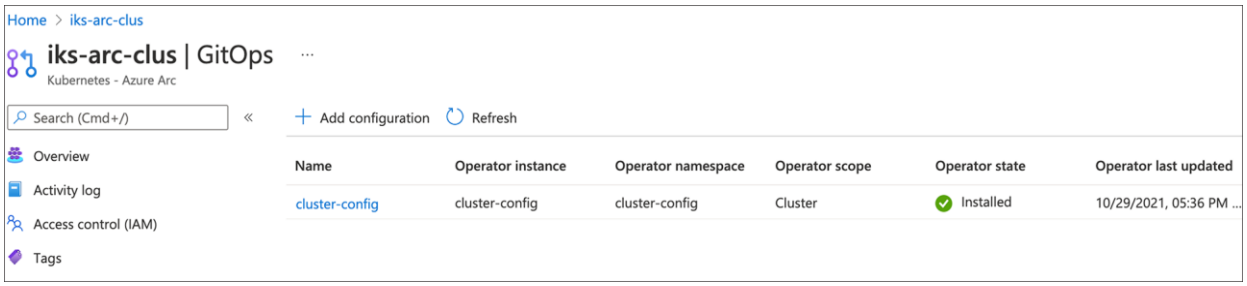


4. Click on "Add Configuration," enter the details, and provide a link to the forked repository containing the manifest files, then Click Add. The scope of the configuration can be applied either to the cluster or to a particular namespace.



© 2022 Cisco and/or its affiliates. All rights reserved.

5. The Azure Arc configuration agent will notice the new configuration applied, connects the cluster to the repository and installs the Flux operator. Click on refresh to see the configuration's operator state change from pending to installed.



6. Verify the namespaces created on the cluster after the configuration is applied

```
> kubectl get ns –show-labels
```

```
snaldurg@DemoVM-VirtualBox:~$ kubectl get ns --show-labels
NAME                      STATUS   AGE     LABELS
azure-arc                 Active   10d     admission.policy.azure.com/ignore=true,app.kubernetes.io/managed-by=Helm,control-plane=true
azuremonitor-containers   Active   10d     <none>
cluster-config            Active   7m7s    <none>
default                   Active   58d     <none>
iks                       Active   58d     <none>
iks-arc-ds                Active   7d22h   <none>
itops                     Active   6m51s   fluxcd.io/sync-gc-mark=sha256.fy_pVXP3osuYHatqaA0yhpTaaGxMbfRnfTpuvK013TY,name=itops
kube-node-lease           Active   58d     <none>
kube-public               Active   58d     <none>
kube-system               Active   58d     <none>
team-a                    Active   6m51s   fluxcd.io/sync-gc-mark=sha256.mMpiUJSoSTcF9YYOtVCZoYQetYEtCP-ogFt5J8Tofo8,name=team-a
team-b                    Active   6m51s   fluxcd.io/sync-gc-mark=sha256.EwaiZQMYdueJ4awJqx3sFUFAJiRs38Ev7q8HNDXVmy8,name=team-b
```

7. Verify the deployment created on the cluster after the configuration is applied.

```
> kubectl -n cluster-config get deploy -o wide
```

```
snaldurg@DemoVM-VirtualBox:~$ kubectl -n cluster-config get deploy -o wide
NAME                    READY   UP-TO-DATE   AVAILABLE   AGE    CONTAINERS   IMAGES                                                SELECTOR
cluster-config          1/1     1            1           8m17s  flux         mcr.microsoft.com/oss/fluxcd/flux:1.21.2              instanceName=cluster-config,name=flux
memcached-cluster-config  1/1   1            1           8m17s  memcached    mcr.microsoft.com/oss/memcached/memcached:1.6.10      name=memcached-cluster-config,namespace=cluster-config
```

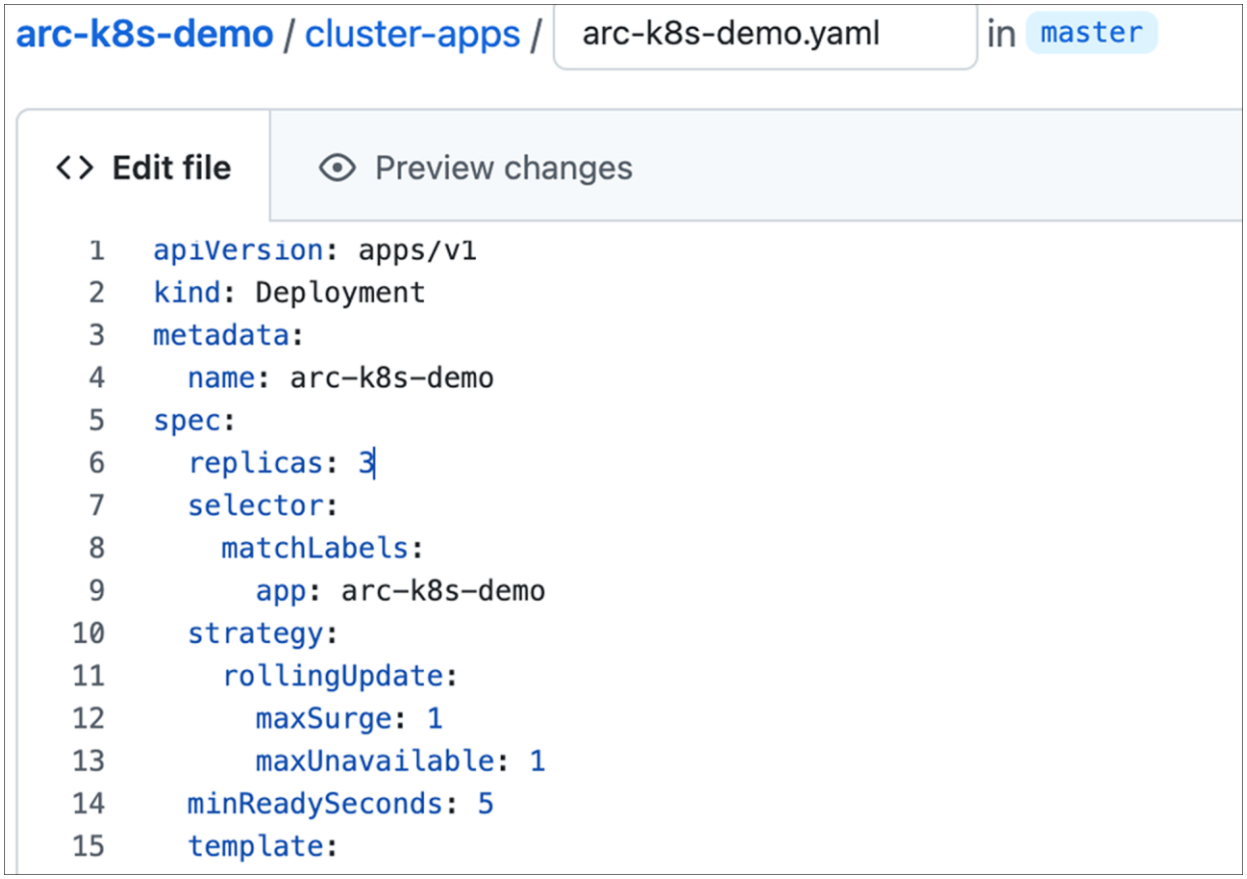
8. Verify the pods created in the default namespace after the configuration is applied. Note a single pod is deployed as per the configuration.

```
> kubectl get pods
```

```
snaldurg@DemoVM-VirtualBox:~$ kubectl get pods
NAME                          READY   STATUS    RESTARTS   AGE
arc-k8s-demo-55db94955-fdws2  1/1     Running   0          144m
```

© 2022 Cisco and/or its affiliates. All rights reserved.

9. Next open the yaml file and change the replica set number from one to three in the repository and commit the changes.



10. After committing the changes, the updates are applied automatically and can be verified using the command below:

```
> kubectl get pods -w
```

```
snaldurg@DemoVM-VirtualBox:~$ kubectl get pods -w
NAME                              READY   STATUS              RESTARTS   AGE
arc-k8s-demo-55db94955-5bmqg   0/1     ContainerCreating   0          3m1s
arc-k8s-demo-55db94955-c7z4z   0/1     ContainerCreating   0          3m1s
arc-k8s-demo-55db94955-fdws2   1/1     Running             0          153m
arc-k8s-demo-55db94955-c7z4z   1/1     Running             0          4m39s
arc-k8s-demo-55db94955-5bmqg   1/1     Running             0          4m53s
```

11. After the update is rolled out successfully, we can see the three replicas of the pod running now.

```
> kubectl get pods
```

```
snaldurg@DemoVM-VirtualBox:~$ kubectl get pods
NAME                              READY   STATUS    RESTARTS   AGE
arc-k8s-demo-55db94955-5bmqg   1/1     Running   0          6m21s
arc-k8s-demo-55db94955-c7z4z   1/1     Running   0          6m21s
arc-k8s-demo-55db94955-fdws2   1/1     Running   0          156m
```

© 2022 Cisco and/or its affiliates. All rights reserved.

Connect Cisco IKS cluster to Azure using Azure CLI

This section shows how to connect a Kubernetes cluster to Azure with Azure Arc-enabled Kubernetes using the Azure CLI.

1. Refer to the above prerequisites section to prepare the environment.
2. Login to the Azure and set the Azure subscription by running the below commands

```
> az login
> az account set –subscription xxxxxxxxxxx
```

3. Run the following command to create a resource group in the Azure:

```
> az group create --name glx-arc-demo --location EastUS --output table
```

```
snaldurg@SNALDURG-M-WH74 ~ % az group create --name glx-arc-demo  --location EastUS --output table
Location    Name
----------  ------------
eastus      glx-arc-demo
```

4. To connect the Kubernetes cluster to Azure.

- Run the following command if the cluster is not behind an outbound proxy server:

```
> az connectedk8s connect --name glx-iks-arc --resource-group glx-arc-demo --location
eastus --tags Datacenter=rtp-glxy owner=snaldurg
```

```
snaldurg@SNALDURG-M-WH74 ~ % az connectedk8s connect --name glx-iks-arc --resource-group glx-arc-demo --location
 eastus --tags Datacenter=rtp-glxy owner=snaldurg
Ensure that you have the latest helm version installed before proceeding.
This operation might take a while...
```

- Run the following command with proxy parameters if the cluster is behind an outbound proxy server:

```
> export HTTP_PROXY=<proxy-server-ip-address>:<port>
> export HTTPS_PROXY=<proxy-server-ip-address>:<port>
> export NO_PROXY=<cluster-apiserver-ip-address>:<port>
> az connectedk8s connect --name <cluster-name> --resource-group <resource-group> --
proxy-https https://<proxy-server-ip-address>:<port> --proxy-http http://<proxy-
server-ip-address>:<port> --proxy-skip-range <excludedIP>,<excludedCIDR> --proxy-cert
<path-to-cert-file>
```

5. To verify the cluster connection run the below command:

```
> az connectedk8s list --resource-group glx-arc-demo --output table
```

```
snaldurg@SNALDURG-M-WH74 ~ % az connectedk8s list --resource-group glx-arc-demo --output table
Name         Location    ResourceGroup
-----------  ----------  ---------------
glx-iks-arc  eastus      glx-arc-demo
```

© 2022 Cisco and/or its affiliates. All rights reserved.

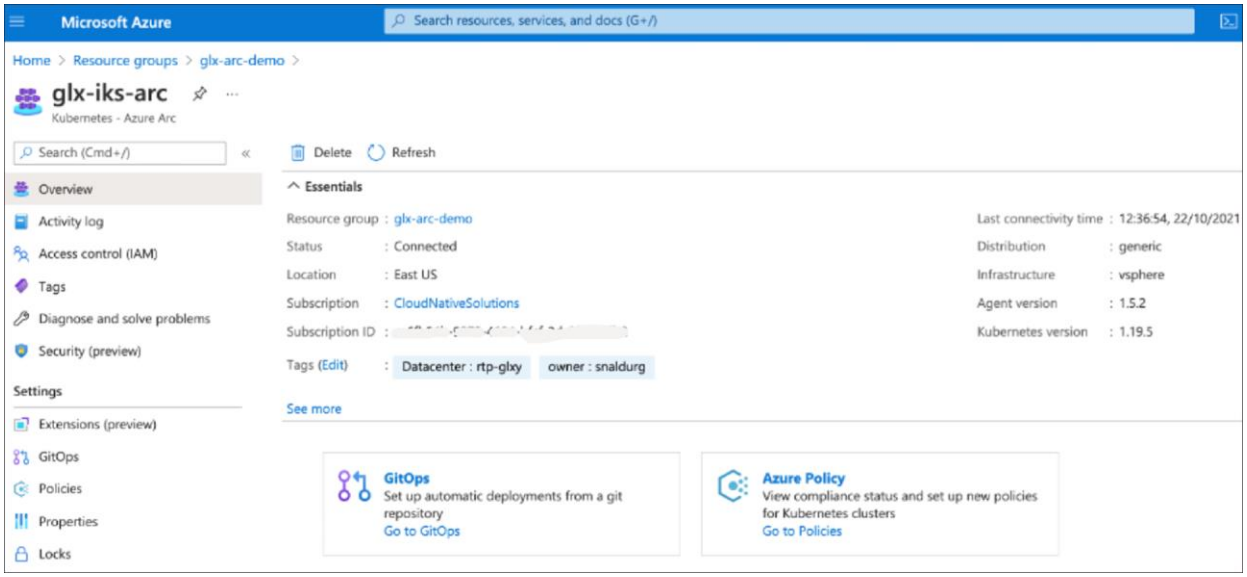
6. To view the Azure Arc agents deployed to the Kubernetes cluster, run the below command:

```
> kubectl get deployments,pods -n azure-arc
```

```
snaldurg@SNALDURG-M-WH74 ~ % kubectl -n azure-arc get pods,deployments
NAME                                            READY   STATUS    RESTARTS   AGE
pod/cluster-metadata-operator-75687dfd87-jmn5x  2/2     Running   0          10m
pod/clusterconnect-agent-689c8cd99-cn7fb        3/3     Running   0          10m
pod/clusteridentityoperator-68676b54db-5nwgq    2/2     Running   0          10m
pod/config-agent-5fd84d4d8b-b9zl6               2/2     Running   0          10m
pod/controller-manager-587d9694c7-stx8d         2/2     Running   0          10m
pod/extension-manager-5b595fbd96-5sm6c          2/2     Running   0          10m
pod/flux-logs-agent-7ff9dc5c99-pkn4h            1/1     Running   0          10m
pod/kube-aad-proxy-747c555884-72mvb             2/2     Running   0          10m
pod/metrics-agent-b5ff69d5c-2z25f               2/2     Running   0          10m
pod/resource-sync-agent-7c77d6f664-w2d8s        2/2     Running   0          10m

NAME                                       READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/cluster-metadata-operator  1/1     1            1           10m
deployment.apps/clusterconnect-agent       1/1     1            1           10m
deployment.apps/clusteridentityoperator    1/1     1            1           10m
deployment.apps/config-agent               1/1     1            1           10m
deployment.apps/controller-manager         1/1     1            1           10m
deployment.apps/extension-manager          1/1     1            1           10m
deployment.apps/flux-logs-agent            1/1     1            1           10m
deployment.apps/kube-aad-proxy             1/1     1            1           10m
deployment.apps/metrics-agent              1/1     1            1           10m
deployment.apps/resource-sync-agent        1/1     1            1           10m
```

7. Log in to the Azure portal and verify that the cluster is listed and in a connected state.



© 2022 Cisco and/or its affiliates. All rights reserved.

For more information

For additional information, see the following resources:

- https://intersight.com/help/saas/resources/intersight_kubernetes_service_user_guide#overview
- https://www.cisco.com/c/en/us/products/collateral/cloud-systems-management/intersight/deploy-kasten-k10-on-cisco-intersight-kubernetes-service-for-container-backup.html
- https://docs.microsoft.com/en-us/azure/azure-arc/kubernetes/overview
- https://docs.microsoft.com/en-us/azure/azure-arc/kubernetes/conceptual-agent-overview
- https://docs.microsoft.com/en-us/azure/azure-arc/kubernetes/conceptual-gitops-flux2

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Printed in USA C11-2784174-00 01/22

© 2022 Cisco and/or its affiliates. All rights reserved.